Representation and Recognition of Uncertain Enemy Policies Using Statistical Models

Robert Suzić
Tel: +46 (0) 8 55 50 32 17
Swedish Defence Research Agency (FOI)
SE-172 90 Stockholm
SWEDEN

Email: robert.suzic@foi.se ; rsu@nada.kth.se

ABSTRACT

In this work we extend from the single agent to the on-line multi-agent stochastic policy recognition problem using a network structure. By using knowledge of agents' interrelations we can create a policy structure that is compatible with that of a hostile military organisation. Using this approach we make use of existing knowledge about the military organisation and thereby strongly reduce the size of the hypothesis space. In this way we are able to bring down the problem complexity to a level that is tractable. Also, by using statistical models in policy recognition we are able to deal with uncertainty in a consistent way. This means that we have achieved improved policy recognition robustness.

*We have developed a **proof of concept** Bayesian Network model. For the information fusion purpose, we show with our model that it is possible to integrate the pre-processed uncertain dynamical sensor data such as the enemy position and combine this knowledge with terrain data and uncertain a priori knowledge such as the doctrine knowledge to infer multi-agent policy in a robust and statistically sound manner.*

1.0 INTRODUCTION

Dealing with uncertain information in a complex and in some cases chaotic environment is the difficult task for military commanders. A set of methods which improve the process of collecting and reasoning about uncertain information is called *information fusion*. The goal of information fusion is to describe a particular state of the world of interest by making best possible use of all available information. In military applications this can apply to anything from the position and type of hostile forces to an enemy's plans and intentions. This paper shows how our knowledge about the enemy can be represented and his policies recognised. We choose Bayesian Networks (BN) as the method for representation of our knowledge about the enemy, designated *static knowledge*, and Dynamic Bayesian Networks (DBN) for inference based on sensor data, designated *dynamic knowledge*. Those problems, as many others, must of course be addressed in the operational system. In order to focus on policy recognition, in this work we do not deal with the classical identification and association problems.

The stochastic nature of policies is derived from the fact that we do not have full knowledge about the enemy and his actions. Instead, policies are represented as discrete probability density functions on different modelling levels. This means that military commanders should not only pay attention to the policy with the highest probability but to all policies that have a significant probability to occur.

Paper presented at the RTO IST Symposium on "Military Data and Information Fusion", held in Prague, Czech Republic, 20-22 October 2003, and published in RTO-MP-IST-040.

Report Documentation Page	*Form Approved OMB No. 0704-0188*

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
00 MAR 2004	**N/A**	**-**

4. TITLE AND SUBTITLE **Representation and Recognition of Uncertain Enemy Policies Using Statistical Models**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Swedish Defence Research Agency (FOI) SE-172 90 Stockholm SWEDEN**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S) 11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM001673, RTO-MP-IST-040, Military Data and Information Fusion (La fusion des informations et de données militaires)., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**29**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

In this paper we will present the dynamical stochastic policy recognition problem. By dynamical we mean that policy can change over time and as soon as new observations arrive. The automatic process of policy recognition is performed for each new observation. The process is on-line and involves many agents (units) acting together. Thus we characterize the task as *on-line multi-agent stochastic policy recognition.*

Situation awareness plays a decisive role in achieving information and next-turn decision superiority on the battlefield. “Battle space is an abstract notion that includes not only the physical geography of a conflict but also the plans, goals, resources and activities of all combatants prior to, and during, a battle and during the activities leading to the battle” [1]. It is generally difficult to derive conclusions about the enemy’s intentions from a chaotic, uncertain and complex environment, see [2]. Military commanders have to be able to act fast. In order to achieve agility military commanders need to have good situation awareness. On-line policy recognition gives users, military commanders in this case, hints about what the enemy is going to do next, provided relevant sensor information and a priori knowledge about the enemy is present. The next step the military commander has to make is to investigate if the warning received from the system is serious. If the system correctly deduces that an attacking policy is the most likely to happen or that the probability is high that the military commander’s own force has been discovered, then situation awareness, i.e., the ability to respond in time, improves and the security level increases by using the on-line multi-agent stochastic policy recognition method.

This work has been performed within the Information Fusion strategic kernel project at the Swedish Defence Research Agency (FOI) and the Royal Institute of Technology (KTH) in Sweden.

2.0 POLICY RECOGNITION AND INFORMATION FUSION

2.1 Aim

On-line multi-agent stochastic policy recognition aims to detect which policies an agent or group of agents are executing by observing the agents' actions and by using *a priori* knowledge about the agents in a noisy environment.

The method chosen for the representation of this task is Bayesian inference using dynamic Bayesian nets and other statistical models. The inference is intended to derive belief measures for enemy plans. Thus, the paper provides yet another example, see also [3, 4, 5, 6], of how Bayesian Networks (also known as probabilistic networks or belief networks) can be used to support situation awareness.

In everyday life people recognise intentions of other people, and in some cases intentions of animals. Here is a burglary example: How can we recognise that a thief or a number of thieves are trying to rob our neighbour’s house? You can be fairly certain that a burglary is taking place if you see that some person is trying to get in through the window of your neighbour’s house. But what if in the next moment some member of your neighbour’s family opens the window to the "burglar"? This means that the probability of a burglary is lowered and you update your belief dynamically (on-line). The hypothesis “an inappropriate visit” gets more support. The probability density function over the hypothesis space changes. This is a main feature in BN modelling, illustrated by similar examples in [10].

In military applications the issue is how to recognise certain military behaviours of the enemy. Using the movement pattern, speed, distance, weather, maneuverability distance to presumptive target, etc., it might be possible to fuse the acquired knowledge about the enemy and use it in policy recognition. The advantage

would be that military commanders, having better knowledge about the enemy's intentions, will be able to act earlier. The ability to act preventively increases as well. In this paper we claim that by using our knowledge about hostile force doctrines and fusing this knowledge with sensor information we can recognise certain military behaviours of the hostile force. Our aim is to demonstrate the statement above in a proof-of-concept model. This work uses a DBN model to represent our beliefs about hostile force units.

2.2 Modeling Approach

As mentioned in the introduction, in this work we do not deal with the classical identification and association problems. A further delimitation is that sensor reports are assumed to be pre-processed so that the input to our model are positions of the enemy represented as discrete probability density functions. In the present work we create a model of a hostile tank company. The model is hierarchical, corresponding to an hierarchical policy structure. The company consists of three tank platoons, each platoon containing three tanks. For each level there is a certain set of *policies* that are invoked by the higher level. The simplest policies, their *atoms*, consist only of a set of *actions*. In this example the simplest policy is the tank (group) policy. More complex policies consist of other policies, also referred to as *sub-policies*, or a mixture of sub-policies and actions. Higher level policies invoke lower level policies down to their action atoms.

The connection between different policy levels is modeled using conditional probabilities which represent uncertain causal relationships. A policy of agent i on level k depends on the policy that a superior agent is executing on level $k + 1$ and on the state of agent i in the previous time step. Single agent stochastic policy recognition is represented by an Abstract Hidden Markov Model (AHMM) in [7]. In Kevin Murphy's Ph.D. work [8] an AHMM is described as a DBN. In this work we extend the on-line single agent stochastic policy recognition problem to the corresponding on-line multi-agent stochastic policy recognition problem by using a network structure that allows many agents. By using knowledge of agents' (hostile units') interrelations we create a policy structure that is compatible with that of a hostile military organisation. Using this approach we both make use of existing knowledge about the military organisation and strongly reduce the size of the hypothesis space. In this way we are able to bring down the problem complexity to a level that is tractable.

There are several advantages with this kind of representation. One of them is that one can build policy hierarchies modularly, in a way that corresponds to military organisational hierarchy. Using this kind of approach we can facilitate:

- the process of knowledge reuse,
- the process of updating knowledge,
- verification and validation processes.

When observing actions of an agent (hostile unit) we face several uncertainties. One kind of these uncertainties are the stochastic outcomes of actions, see [7, 9]. An agent intends to carry out some actions to achieve his goals but he is in some cases prevented from using these actions. In some situations, the effects of the agent's actions therefore do not correspond to the goal state. E.g., in some cases one tank unit is planning to take a hill but during some part of the plan execution it is understood that this is not possible because of bad maneuverability. In our model we do not care if the agent is rational. Instead we model and analyse if some policies that the agent is executing are more likely than others. This modelling approach respects terrain data, a list of the agent's possible targets and our knowledge about the agent's behaviour. Other obstacles in the policy recognition problem are uncertain observations and incomplete knowledge about the agent and his behaviour.

In order to be able to handle those different types of uncertainties we use Bayesian Networks as the modelling technique. See [8, 10, 11] for an introduction to BN and DBN. A Bayesian Network is a set of uncertain causal connections between variables of interest. The connections are modeled as conditional probability distributions. A BN uses a distributed representation of the current state. Thus we can also represent the probability density function in a distributed way. There are three types of variables in our type of BN:

1) hypothesis variables,
2) information or evidence variables,
3) mediating variables.

In our case, the hypothesis variables are the agent's policies. Hypothesis variables are either impossible or too costly to observe. Information or evidence variables represent our observation models, e. g. the uncertain position of the enemy. Mediating variables are introduced for special purposes. In this case we use mediating variables to model terrain restrictions and the agent's doctrines.

The multi-agent policy hierarchy in this paper consists of the policy of the company at the top level, the platoon policies at the next level and the tank policies. Our BN modelling approach is that the company policy causes change in platoon policies and platoon policies cause change in group (tank) policies. One reason for using this modelling approach is that this model follows military hierarchy; commanders give orders to their subordinates who are superior at the next level. The other reason is to minimize complexity, see also [7, 13]. The ability to represent uncertain causal relationships is one of the main advantages of Bayesian Networks. A BN can only represent a snapshot of the situation, whereas a DBN allows new evidence to arrive while taking into account old variable values.

Evidence nodes propagate new information through the network by updating variables of the DBN.

2.3 Model Description

Our model was built by using both a Bayesian Network and a Dynamic Bayesian Network representation. The variables in the BN graph are represented by nodes while uncertain causal relationships are represented by arcs. The description of uncertain causal relationships is performed by using conditional dependence probabilities between the variables. By using BN we are able to express our subjective belief. As an example, we can state that one type of unit formation has lower probability to occur when maneuverability is bad, see also "The Movement-Analysis Challenge Problem" in [1].

In our model we make a distinction between hypothesis nodes (policies nodes, discovery node), mediating nodes (restriction nodes such as the terrain, doctrinal nodes such as formation) and observation nodes (sensor nodes).

Policy hierarchy is represented by the BN. Policy for each agent (hostile unit) is represented as a BN node. The simplest policy is on tank (group) level, $k = 0$.

Tank i's policy variable, $\pi_{o,i}$, has the following discrete states:
$\pi_{o,i}$:= <agent i is moving in the direction of our own force, agent i is moving in the direction opposite to our own force, agent i is moving in neutral direction>
On the next level we have the policy of the tank platoon i at level $k = 1$:
$\pi_{1,i}$:= <attack, defence, reconnaissance, march>
Finally on the top level we have the policy of the tank company at level $k = 2$:
$\pi_{2,i}$:= <frontal attack, frontal attack and flange attack, defence, delay battle, march>

In Figure 1 we show a Bayesian network representing the policy hierarchy model of a hostile company.

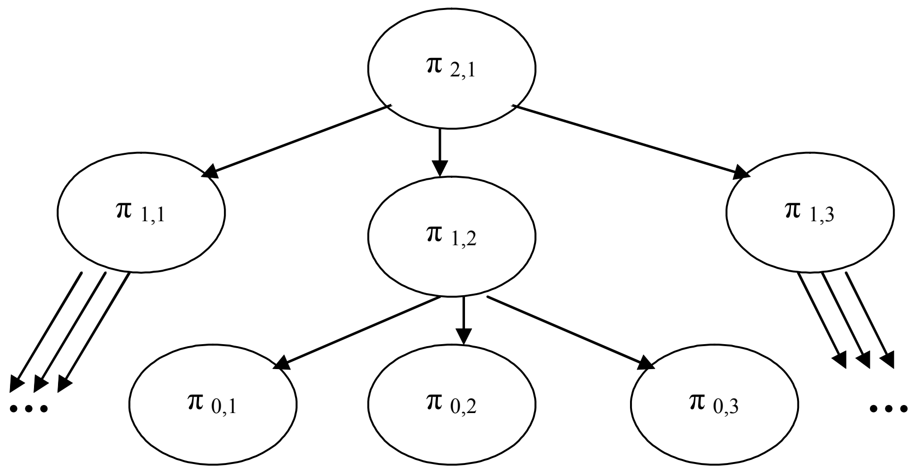


Figure 1: Policy hierarchy.

In this model we combine terrain representation with uncertainty about hostile force position. The output of this fusion is represented in a BN node called maneuverability. We find it important to model information about the terrain in the policy recognition problem. In particular environments, particular policies are assumed to be more probable than in some other environments. In our DBN model the representation of environment is limited to terrain, visibility, and the possibility to find cover. The variables formation, distance to presumptive target, and direction of guns are used to connect observations to different policies. We call those nodes strictly doctrinal. The hostile company model is implemented in MATLAB using K. Murphy's BN package [8,16]. The DBN representation of a hostile company is visualised in Figure 2.

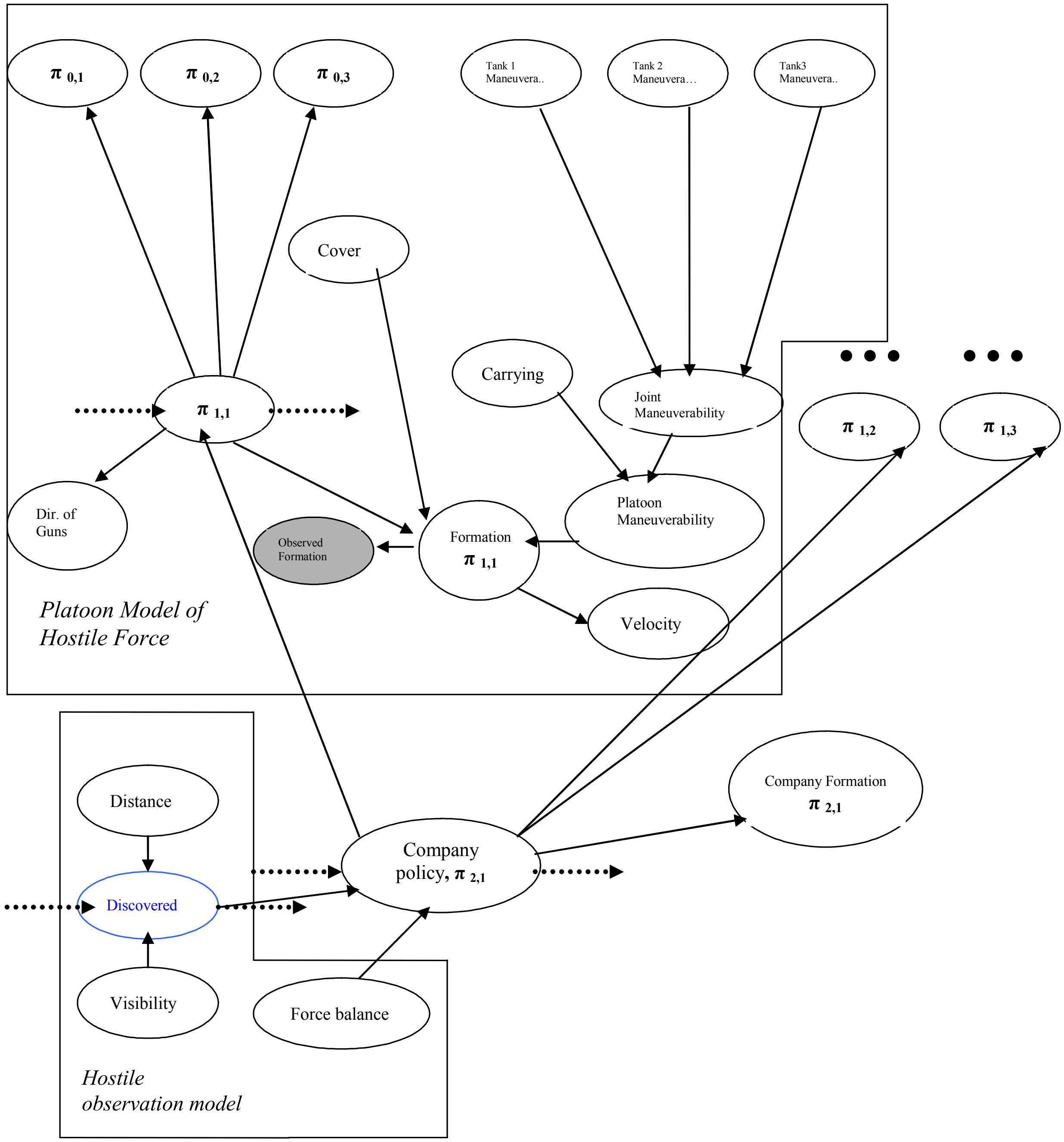


Figure 2: Bayesian Network representing a hostile company.

The whole network contains 57 nodes. The purpose of the network is to make qualified estimation of the opponent's behaviour based on observations, knowledge about opponent's doctrines as well as data from the terrain. If the whole terrain would be represented as Bayesian nodes then the problem of high computational complexity would appear. To represent the terrain in this model we use a fragmented representation of it. Such

nodes as *Tank Maneuverability*, *Carrying* and *Cover* represent terrain. The probability distribution of the value *Tank Maneuverability* depends on many factors. For instance, the tank maneuverability depends on slopes in the surroundings and if there are any significant obstacles such as large water areas. It depends also on if the tank is on road or not. In this implementation we only use this fact when representing tank manoeuvrability. The carrying capacity depends on the terrain's capability to carry heavy loads such as tanks. The *Cover* depends, in our implementation, on vegetation. In a pre-processing stage, before the simulation is started, we performed a classification of the terrain for each pixel of the map that we had read into MATLAB. That is, the classification of each pixel is based on whether it represents a road or not. In the same manner we performed classification on vegetation of the terrain. As the result we get data that represents different aspects of the terrain.

We use MATLAB functions which take an estimate of the tank position as input and make a weighted calculation reading the terrain data that corresponds to the tank position and its close surrounding. The weighting function should correspond to the uncertainty of the observation. The greater the uncertainty about enemy tank position, the more data representing the surroundings should be taken into concern. If an observation is pretty certain then weighting should be high on data representing an estimate of the tank position. In our example we use only rectangular distributions over a set of pixels. The output of the functions are probability distributions. Those values are entered as soft evidence in Bayesian nodes. The example of such nodes are *Cover* and *Tank Maneuverability*. The variable *Tank Manoeuvrability* has two states, Good and Bad. A MATLAB function assigns the probabilities over variable *Tank Maneuverability* states for each observation.

To facilitate the defining of our a priori knowledge of the variable *Platoon Maneuverability* we introduce a mediating node designated as *Joint Maneuverability.* Consequently, by using the mediating node the variable *Platoon Maneuverability* gets two parents instead of four. Thus we define distributions that depend on two parents instead of four. This method is called *divorcing*, see also [10] p. 52.

One of the key variables that reveal enemy behaviour and intentions is the enemy formation. It is represented as a Bayesian node in the network. According to doctrine manuals, when the enemy has the intention to attack it usually attacks in battle line formation. When transporting to a certain destination the enemy transports in march formation. There are many reasons why the enemy may not use a certain type of formation. One of the factors that have influence on building a formation is the environment. When the platoon maneuverability is bad the enemy will not have the opportunity to attack in battle line and the probability of the formation type battle triangle increases. The probability that the enemy is performing reconnaissance increases when the enemy is moving in a stepped formation. As for all Bayesian variables in the network, we define a priori knowledge about the variable *Formation* as the probability density function **P**(*Formation* | *Platoon Policy, Cover, Platoon Maneuverability*). In Table 1 we describe the distribution that we have implemented. We found the variable *Formation* important because it models the connection between doctrines and environment in a statistical and therefore soft manner.

Table 1: A Priori Distribution of the Formation variable

P (Formation | Platoon Policy, Cover, Platoon Maneuverability)						
Policy Platoon	**Cover**	**Maneuverability**	Formation value = Lead	Formation value = Battle Line	Formation value = Stepped Formation	Formation value = Battle Triangle
1)Attack	**Good**	**Good**	**0.05**	**0.75**	**0.15**	**0.05**
2)Defend	Good	Good	0.05	0.60	0.3	0.05
3)Reconnaissance	Good	Good	0.35	0.2	0.25	0.2
4)March	Good	Good	0.9	0.01	0.01	0.08
5)Attack	Bad	Good	0.01	0.79	0.1	0.1
6)Defend	Bad	Good	0.01	0.53	0.41	0.05
7)Reconnaissance	Bad	Good	0.3	0.3	0.35	0.05
8)March	Bad	Good	0.7	0.2	0.08	0.02
9)Attack	**Good**	**Bad**	**0.08**	**0.5**	**0.25**	**0.17**
10)Defend	Good	Bad	0.08	0.55	0.35	0.02
11)Reconnaissance	Good	Bad	0.3	0.13	0.47	0.1
12)March	Good	Bad	0.75	0.01	0.04	0.2
13)Attack	Bad	Bad	0.08	0.3	0.2	0.42
14)Defend	Bad	Bad	0.02	0.2	0.5	0.28
15)Reconnaissance	Bad	Bad	0.2	0.38	0.4	0.02
16)March	Bad	Bad	0.56	0.08	0.12	0.24

In the first row where *Platoon Policy* = Attack, *Cover* = Good and *Manoeuvrability* = Good, the probability for *Formation* = Battle Line is 75 % while the hypothesis *Formation* = Battle triangle has only 5 % support. However, if the maneuverability is bad the enemy may not be able to hold the battle line formation. Therefore if *Maneuverability* = Bad then the triangle formation gets greater support, 17 %, see row nine. This is easy to explain by the fact that the enemy will not have good coordination ability and enough free space to attack in the battle line formation when manoeuvrability is bad.

In the same manner we define knowledge that the most probable formation is Lead when the platoon is performing march. The lead formation is also under influence of the terrain and the cover. Therefore probability for *Formation* = Lead decreases from 90 % to 75 % when maneuverability changes from Good to Bad. If *Maneuverability* = Bad and *Cover* = Bad this probability would decrease further to 56 %.

A natural question that arises is: Why these numbers? The simple answer is that we model this distribution to make a distinction regarding which hypothesis is more probable. The purpose was not to define, and we doubt that is possible in this application area, how probable a certain hypothesis is. The numbers can differ. E. g. in the previous example, row three in Table 1, the a priori support for the hypothesis Battle Line could be redefined and set to 70 %. At the same time the support for the remaining hypotheses would increase. However, the qualitative representation should remain the same considering the knowledge gained from doctrine manuals. That means that the shape of the probability density function should remain the same. In this example, row three, the support for the hypothesis Battle Line should be higher than the support for any other hypothesis. However, how much higher this support should be is an open issue. This kind of approach is part of the Bayesian way to solve statistical problems. Instead of expressing for which confidence interval the certain hypotheses is valid, we compare hypotheses as in this example. The output in this case is

the distribution of the different hypothesis variables such as policies for platoon, company policy and our guess if the enemy has discovered us. In the same manner as we modeled the input data, attention is focused on which hypotheses are the most probable to occur given the a priori knowledge and sensor reports. Second priority, but also of importance, is their quantitative representation. E. g. we show in section three tables that represent the most probable and the least probable hypotheses for different time steps.

According to our model, the variable *Observed Formation* depends on the actual formation.

Because of the environment, uncertain observations and enemy's coordination problems we are not always able to observe his formation pattern. It is the rule rather than the exception that the enemy's formations do not follow the same geometrical properties as described in the doctrine manuals. Therefore we introduced a heuristic function in MATLAB that takes the estimates of the tank positions as inputs and as output returns the distribution of the observed formation's values. The result is entered as soft evidence in the variable *Observed Formation*. By Bayes rule it has influence on the value *Formation*.

The node *Force Balance* represents status on the battle field. If the enemy is stronger the probability that he will attack is higher than if he is weaker.

The company policy is also under strong influence of discovery. To trigger attack behaviour the enemy has first to discover our forces. Therefore we also build a limited model that represents hostile force ability to observe us. This model is a part of the network. We assume that the discovery depends only on the distance from the enemy to our forces and visibility.

Another aspect of the network representation is that some node values are under influence of the node values of the previous time step. The BN gets dynamic properties. This kind of the network is denoted as DBN. The topology of the network remains the same but the node values vary with the time. The distribution of the variable is conditioned by its parent(s) and the node values of the previous time step.

The list of nodes that are time dependent is the following: *Platoon policies*, *Company policy* and *Discovered*. The reason for introducing those variables as time dependent is purely logical. If enemy has discovered us in one time step, $t - 1$, it implies that the probability that he rediscovers us, finds us again, is higher in the next time step, t. The similar property follows company and platoon policy.

We define a conditional probability density function for the *Discovered* node in Table 2.

Table 2: A priori Distribution of the variable Discover

P(Discovered (t)| Discovered(t-1), Distance(t),Visibility(t))				
Discovered (t-1)	**Distance(t)**	**Visibility(t)**	Discovered = Yes	Discovered = No
Yes	Near	Good	0.99	0.01
No	Near	Good	0.8	0.2
Yes	Neutral	Good	0.9	0.1
No	Neutral	Good	0.65	0.35
Yes	Far away	Good	0.8	0.2
No	Far away	Good	0.4	0.6
Yes	Near	Bad	0.9	0.1
No	Near	Bad	0.7	0.3
Yes	Neutral	Bad	0.8	0.2
No	Neutral	Bad	0.4	0.6
Yes	Far away	Bad	0.6	0.4
No	Far away	Bad	0.15	0.8

In the order to achieve tactical superiority on the battlefield, tanks maneuver very often. That implies for our modeling approach, that we do not connect nodes representing tank policies over time. We connect platoon policies and company policy over time because of higher inertia. If the whole company is attacking at one time step there is significant probability that the company will continue to execute its policy Attack in the next time step.

To variable *Company formation* we assigned abstract type values. We found it difficult to make automatic formation recognition in a manner that the doctrine manual describe formation patterns for company formation. Instead we defined values for the company formation as follows: Lead, Together, Spread, and Very Spread. Again, we use a MATLAB function that reads the estimates of all enemy tank's positions and assigns a value to *Company formation*.

3.0 SCENARIOS, SIMULATION AND RESULTS

We have visualised the movement of a hostile company unit. The hostile company unit consists of three platoons. Each platoon consists of three tanks. There are two scenarios that we use for this simulation. The first one is an attacking scenario and the second one is a movement, march, scenario. The first phase of the both scenarios is the same. In the beginning, the tanks are marching in a neutral direction. The company is not spread over a vast area and the distance to our forces is large. MATLAB functions take as input enemy's tank positions, position of our forces and terrain data. The outputs of those functions are the probability distributions over the variables *Cover, Distance, Tank Policy,* (Platoon) *Formation, Company Formation* and *Tank Maneuverability*. This operation is performed for each tank. The uncertainty about the observations takes into account how the information of the pixels of interest should be weighted. In our example we used a simple approximation of the rectangular distribution over each position.

In time step two observations about current positions and direction of the enemy arrive, see Figure 3. We performed the computation on-line and the results for this step are documented in Table 3.

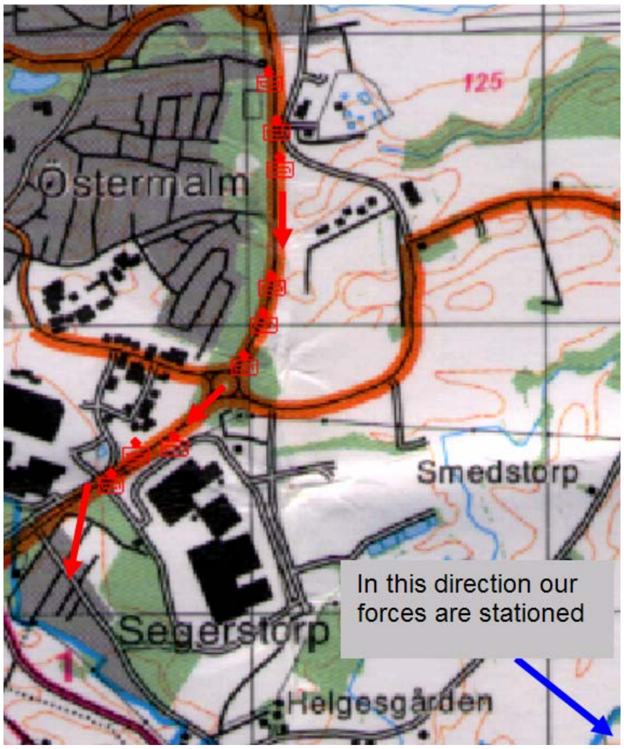


Figure 3: The enemy company is approaching at time step two.

Table 3: Company, platoon policies and values

Most probable and least probable policies at time step two (Figure 3)		
	Most probable states *(Probability %)*	**Least probable *(Probability %)***
Company Policy	March *(47 %)* Flange attack *(21 %)*	Defence *(3 %)*
Platoon One Policy	March *(80 %)*	Defence *(5 %)*
Platoon Two Policy	March *(70 %)*	Defence *(7 %)* Attack *(8 %)*
Platoon Three Policy	Attack *(90 %)*	Defence *(0.5 %)* March *(1 %)*

The probability that the enemy company has discovered us is 33 %. The most probable state of platoon three, according to Table 3, is Attack. The explanation is that this platoon is approaching in the direction towards us. However, the most probable state of the company policy is March. This is achieved by weighting with other nodes including the two other platoon policies. It is usually difficult to infer intentions of a single tank if this unit is not put in a greater context such as platoon or company.

Figure 4: The enemy company is passing by.

After some time observations are received. The enemy begins to approach and then passes by, see Figure 4. The movement and formation pattern indicates that enemy has not discovered us although the distance is short. Thus, the most probable hypothesis is that the hostile force is performing march with probability 98 %. The most probable platoon two policy is March and is 97 % in this case. But for platoon three there is a probability of 44 % that the platoon will attack us and only 27 % probability that this platoon is marching.

The probability that the enemy company has discovered us in this time step (time step = 6) is only 7 %.

3.1 Attacking Scenario

At this time step we divided the scenario in two. The first one is the attacking scenario.

The enemy forces move toward us in two line formations from two different directions.

Figure 5: The enemy approaches in attacking formation.

The first and the third platoon change direction to us and begin approaching. The second platoon continues to move in the same, neutral, direction and follows the road, see Figure 5. We obtain the following results and show them in Table 4.

Table 4: The results for time step seven

Most probable and least probable policies at time step seven in attacking scenario (see Figure 7)		
	Most probable states *(Probability %)*	**Least probable *(Probability%)***
Company Policy	Flange attack *(48 %)* Delay Battle *(25 %)* Frontal *(23 %)*	Defence *(0.5 %)* March *(3 %)*
Platoon One Policy	Attack *(96 %)* Reconnaissance *(3%)*	March *(0.3 %)* Defence *(0.7 %)*
Platoon Two Policy	March *(92 %)* Reconnaissance *(6 %)*	Attack *(1,4 %)* Defence *(1,6)*
Platoon Three Policy	Attack *(61%)* Reconnaissance *(25 %)* Defence *(9 %)*	March *(5 %)*

The formation of the company is much spread out and the most probable type of attack is flange attack combined with frontal attack. On platoon level the platoon two seems to still perform march while the other two platoons have attacking policy as the most likely behaviour.

The probability that we have been discovered is increased to 51 %.

Figure 6: The company movement towards us at time step seven.

In the next time step, eight, observations arrive. The movement pattern is towards us and the formations of all the platoons are battle lines. The probability of a frontal attack increases to 54 % and the probability of a combined frontal and flange attack decreases to 43 %. This means that the probability that the company will attack is extremely high. If we add probabilities of both attacking policies it is 97 %. The probability that we have been discovered increases further to 67 %.

3.2 Marching Scenario

At time step six the enemy will continue the march without performing any attacking movements, see Figure 5. The tank positions are the same as in the attacking scenario but the direction is different. The enemy force will follow the road.

Therefore the probabilities of each hypothesis will be different.

Table 5: Marching enemy

Most probable and least probable policies at time step seven in marching scenario (see Figure 5)		
	Most probable states *(Probability %)*	**Least probable *(Probability %)***
Company Policy	March *(90 %)* Delay Battle *(6 %)*	Attack *(0.1 %)*
Platoon One Policy	March *(70 %)* Reconnaissance *(23%)*	Attack *(0.09 %)*
Platoon Two Policy	March *(97 %)*	Attack *(0,04 %)*
Platoon Three Policy	March *(96 %)*	Attack *(0,08%)*

On both company and platoon levels there is strong support for the hypothesis value March, see Table 5. The probability of attack is low. There is only weak support for the hypothesis Delay Battle. The probability that we have been discovered drops to 10 %.

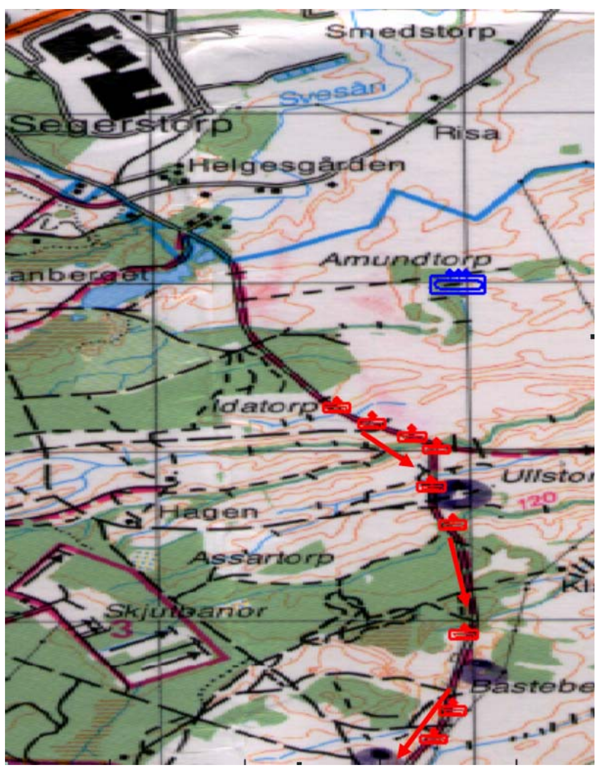


Figure 7: The company movement running away at time step eight marching scenario.

The support for the hypothesis March will increase when the enemy company continues to march in the direction opposite to our position. In Figure 7, we give an example of how the enemy force is continuing its

march and how this hypothesis gets even more support. The probability for the company performing march is increased to 98 % while the probability that we have been discovered drops further to 7 %.

4.0 CONTRIBUTIONS AND CONCLUSIONS

One of the difficult tasks in plan recognition is the interaction between the terrain and recognition processes, see [12]. To decrease complexity we do not use terrain representation by Bayesian nodes. The terrain is instead represented in different data sets. When new observations arrive the MATLAB functions combine them with the relevant terrain data and supply terrain nodes with soft evidence. This information is propagated through the network as the statistical inference process.

The policies do not have stopping conditions in the same manner as in the policy recognition problem where single agents are moving through different rooms, see [7]. In [7], there are distinct sets of policies for each room. The stopping conditions are the walls of the rooms. In the military domain it is usually hard to find such boundaries, but you can connect certain terrain types and certain military policies.

By using statistical models in plan recognition we are able to deal with uncertainty in a consistent way. This means that we have achieved improved policy recognition robustness. In the paper [14], a correlation formula between observations and objects is presented but the drawback is that the plan recognition presented does not take different abstraction levels into account.

We state that our model also has a vague qualitative interpretation. The structure of nodes and arcs explains the model in a qualitative way, see [15]. Also, the results may be interpreted comparing hypotheses instead of expressing how probable they are. DBN are used in this implementation by representing our knowledge in a fragmented manner. By using this kind of approach we obtain a better overview. The knowledge is transparent and the black box concept is avoided. Our model is still incomplete in the sense that we do not incorporate the association and identification problems.

One of the main challenges in this work was to model and extend the problem structure from previously studied single-agent policy recognition to multi-agent policy recognition that is applicable in military information fusion. This was achieved by combing several modelling approaches. The first issue was to model agents and interrelations between different agents on different abstraction levels in the DBN model. By using DBN we also connect policies over time. The second issue was to represent the relevant terrain for the DBN model in a fragmented and dynamical manner for each new observation. Finally, we implemented functions for recognition of the physical relations between the agents such as formations and enemy tank policies. Our simulation results show that it is possible to recognise certain multi-agent stochastic policies on-line.

5.0 ACKNOWLEDGMENTS

I would like to express my thankfulness to: Stefan Arnborg (KTH), Per Svensson (FOI), Magnus Sparf (FOI), Claes Henrikson (FM), Ronnie Johansson (KTH), Joel Brynielsson (KTH), Klas Wallenius (SAAB), Karim Oukbir (KTH), Jesper Fredriksson (KTH), Pontus Svensson (FOI), Hedvig Sidenblad (FOI), Fredrik Samulsson (KTH/FOI) and Mateusz Moszkowicz (SU).

6.0 REFERENCES

[1] Paul Cohen, Robert Schrag, Eric Jones, Adam Pease, Albert Lin, Barbara Starr, David Gunning and Murray Burke, "The DARPA High-Performance Knowledge Bases Project", AI Magazine, Winter 1998, p. 30.

[2] Pearce, A. R. and Heinze, C. A. and Goss, S. "Enabling perception for plan recognition in multi-agent air-mission simulations". Proc. Fourth International Conference on Multi-Agent Systems (ICMASS2000), pp 427-428, 2000.

[3] K.B. Laskey, S.M. Mahoney and E. Wright "Hypothesis Management in Situation-Specific Network Construction", Proc. Seventeenth Conference on Uncertainty in Artificial Intelligence, 2001.

[4] S. Mahoney and K.B. Laskey "Constructing Situation Specific Networks", Proc. Fourteenth Conference on Uncertainty in Artificial Intelligence, 1998.

[5] G. Laskey, K. Laskey "Combat Identification with Bayesian Networks", Proc. Command and Control Research and Technology Symposium 2002.

[6] A.J. Pfeffer "Probabilistic Reasoning for Complex Systems" (Ch. 8.2, p. 257), Stanford University, March 1999, USA.

[7] Bui, H. H., Venkatesh, S., and West, G. "Policy recognition in the Abstract Hidden Markov Model", Journal of Artificial Intelligence Research, 17: 451-499, 2002.

[8] Kevin P. Murphy, "Dynamic Bayesian Networks: Representation, Inference and Learning", University of California, Berkeley, 2002, USA.

[9] Jim Blythe, "Decision-Theoretic Planning", AI Magazine, Summer 1999.

[10] Finn V. Jensen, "An introduction to Bayesian Networks", UCL press, London, ISBN: 1-85728-332-5, 1996.

[11] Bruce D'Ambrosio, "Inference in Bayesian Networks", AI Magazine, p. 37-54.

[12] Robert P. Goldman, Christopher W. Geib and Christopher A. Miller, "A New Model of Plan Recognition", Proc. Fifteenth Conference on Uncertainty in Artificial Intelligence, 1999.

[13] Terran Lane and Leslie Pack Kaelbling, "Toward Hierarchical Decomposition for Planning in Uncertain Environments" , MIT Artificial Intelligence Laboratory, May 2000, USA.

[14] Frank Mulder, Frans Voorbraak, "A formal description of tactical plan recognition", Information Fusion 4 (2003) 47-61.

[15] Jon Doyle and Richmond , "Background to Qualitative Decision Theory", AI Magazine, American Association for Artificial Intelligence, Summer 1999, USA.

[16] Kevin Murphy, "Bayes Net Toolbox for Matlab" , *Last updated on 29 May 2003*, http://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html

Representation and Recognition of Uncertain Enemy Policies Using Statistical Models

Robert Suzic
Swedish Defense Research Agency (FOI)

robert.suzic@foi.se;
rsu@nada.kth.se

Background

- It is generally difficult to derive conclusions about the enemy's state from a chaotic, uncertain and complex environment
- To achieve agility military commanders need to have good situation awareness
- On-line policy recognition gives users, military commanders in this case, hints about what the enemy is going to do next, provided relevant sensor information and a priori knowledge about the enemy

Aim

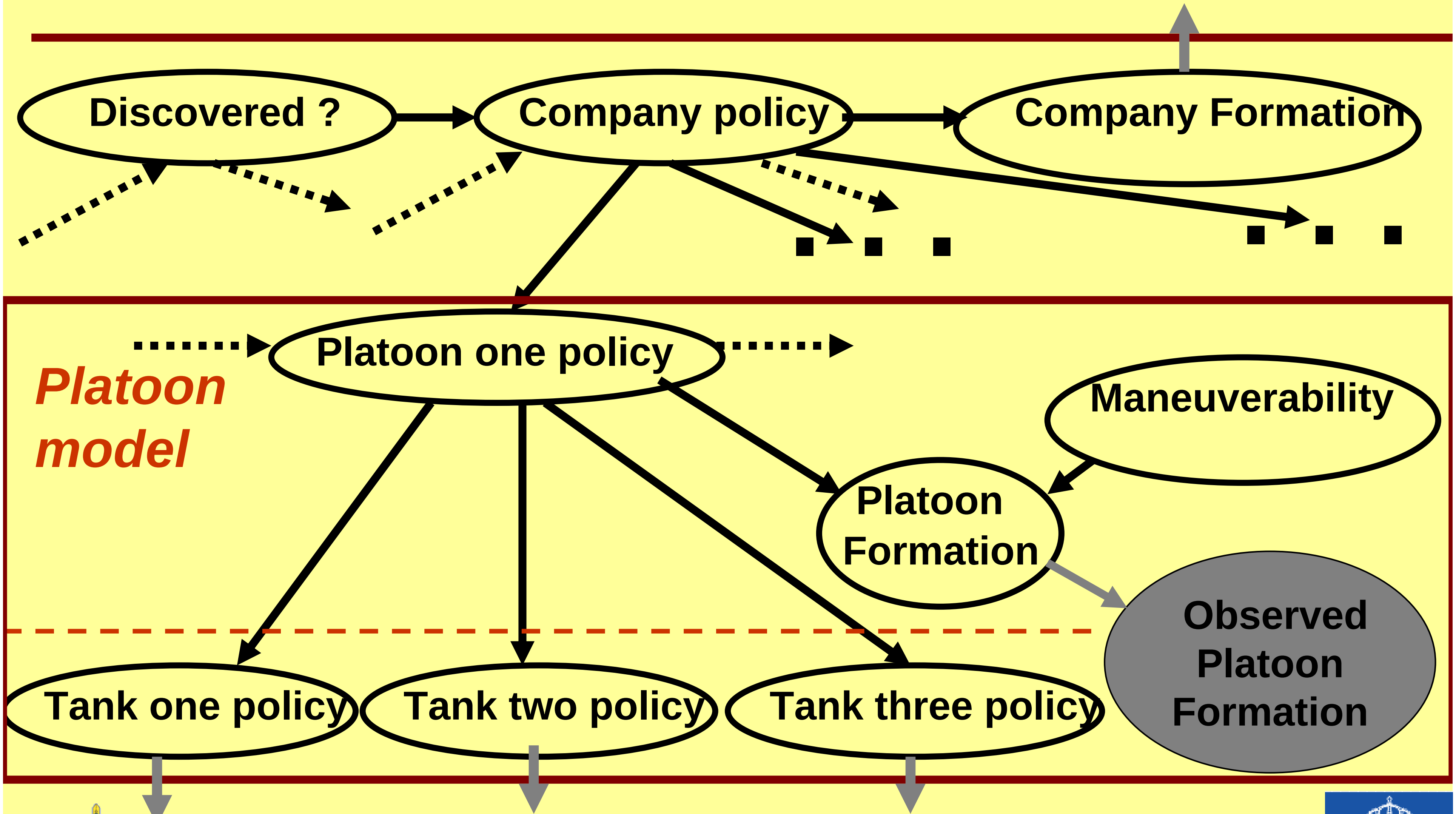
- This paper shows how our knowledge about hostile units can be represented and used for the on-line stochastic multi-agent policy recognition.

- Policy recognition is hierarchical and policy recognition is performed at each level and belief is propagated to the next level.

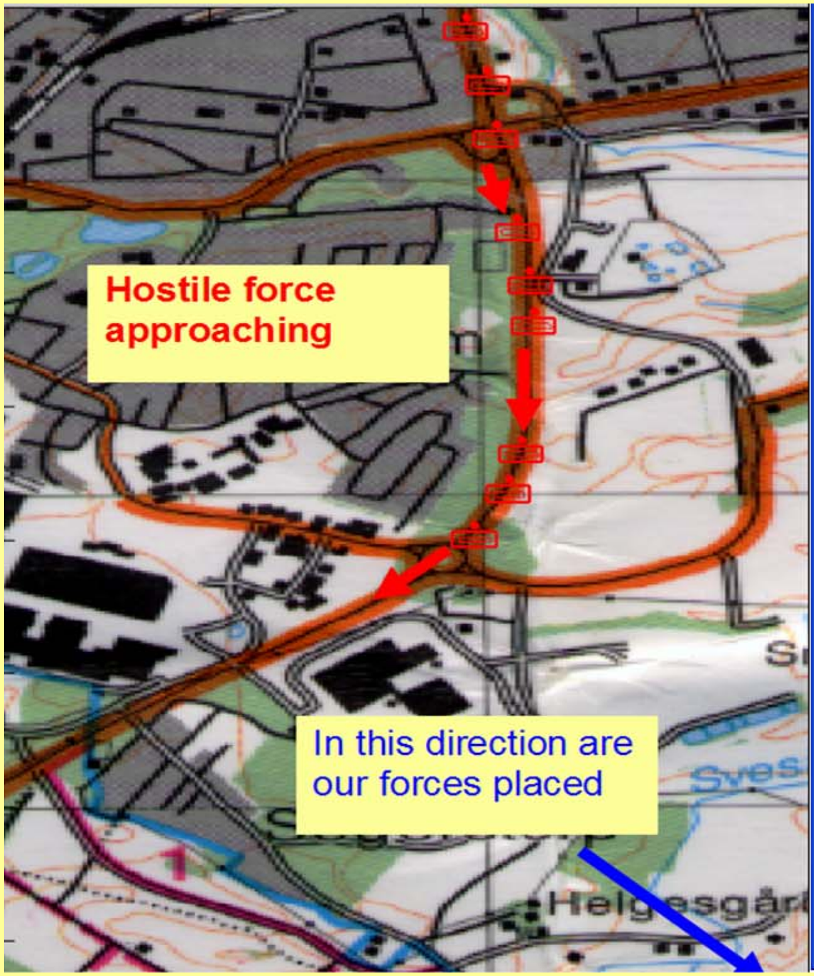
Modeling Approach

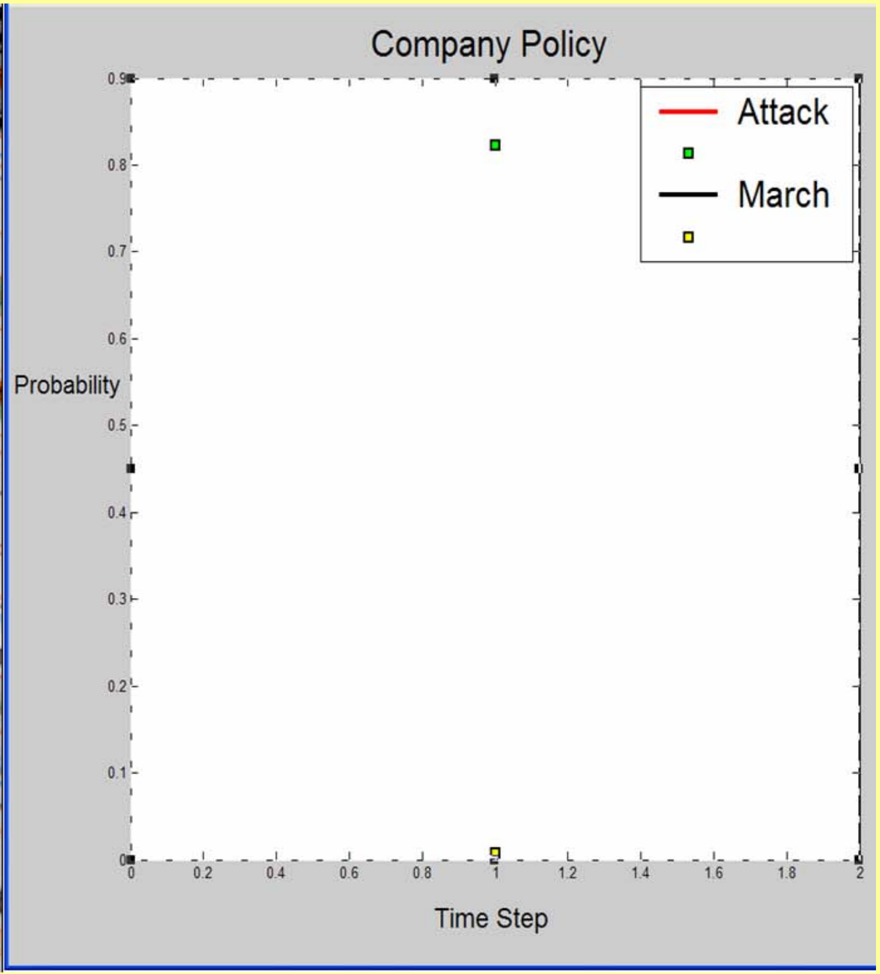
- We model and analyse if some policies that the agent is executing are more likely than others
- Dynamic Bayesian Networks (DBN)
- Heuristic functions that supply the network in form of soft evidence
- To represent the terrain in this model we use a fragmented representation of it.

Tank Company Model: Dynamic Bayesian Network



Results

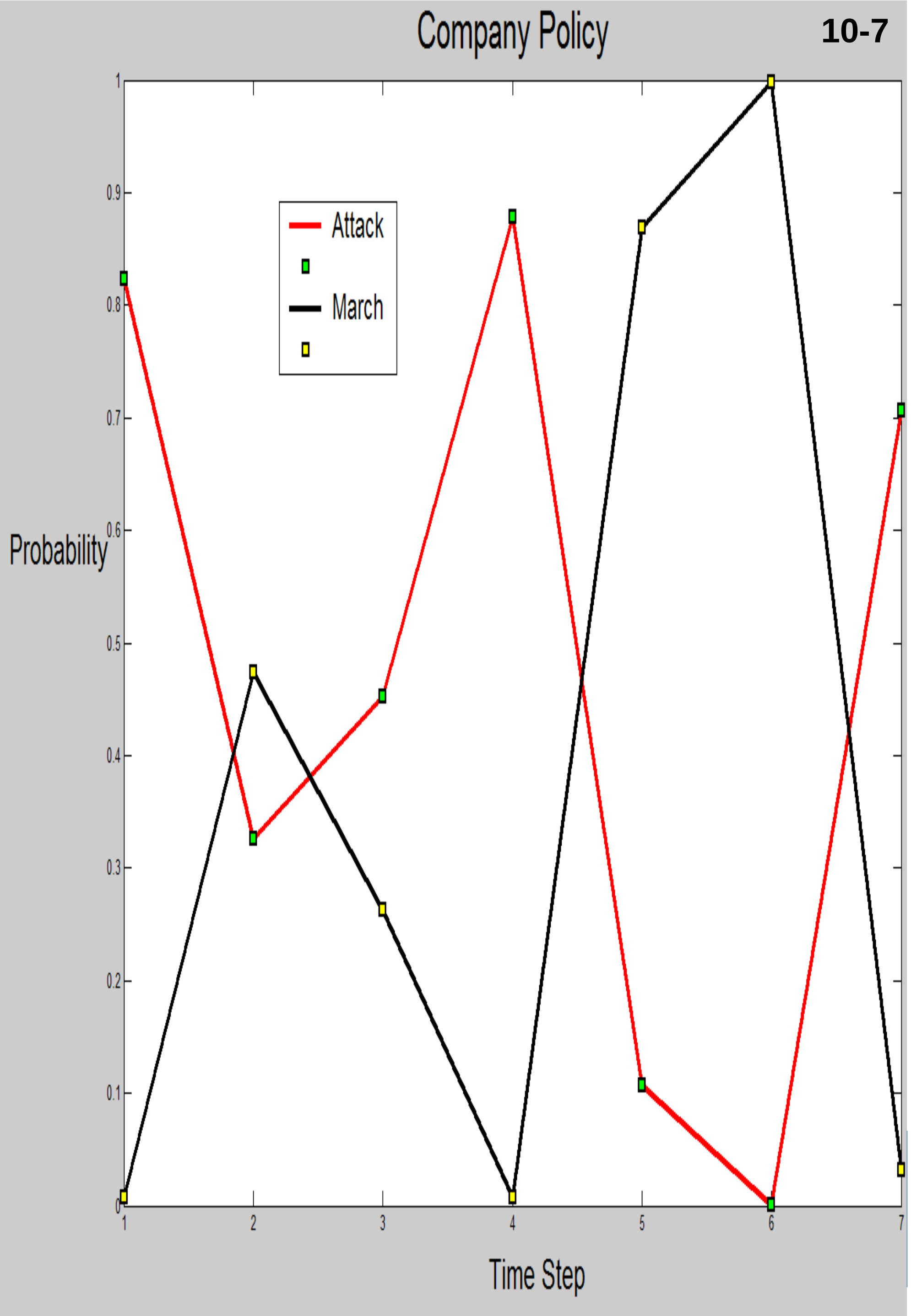




Results.exe

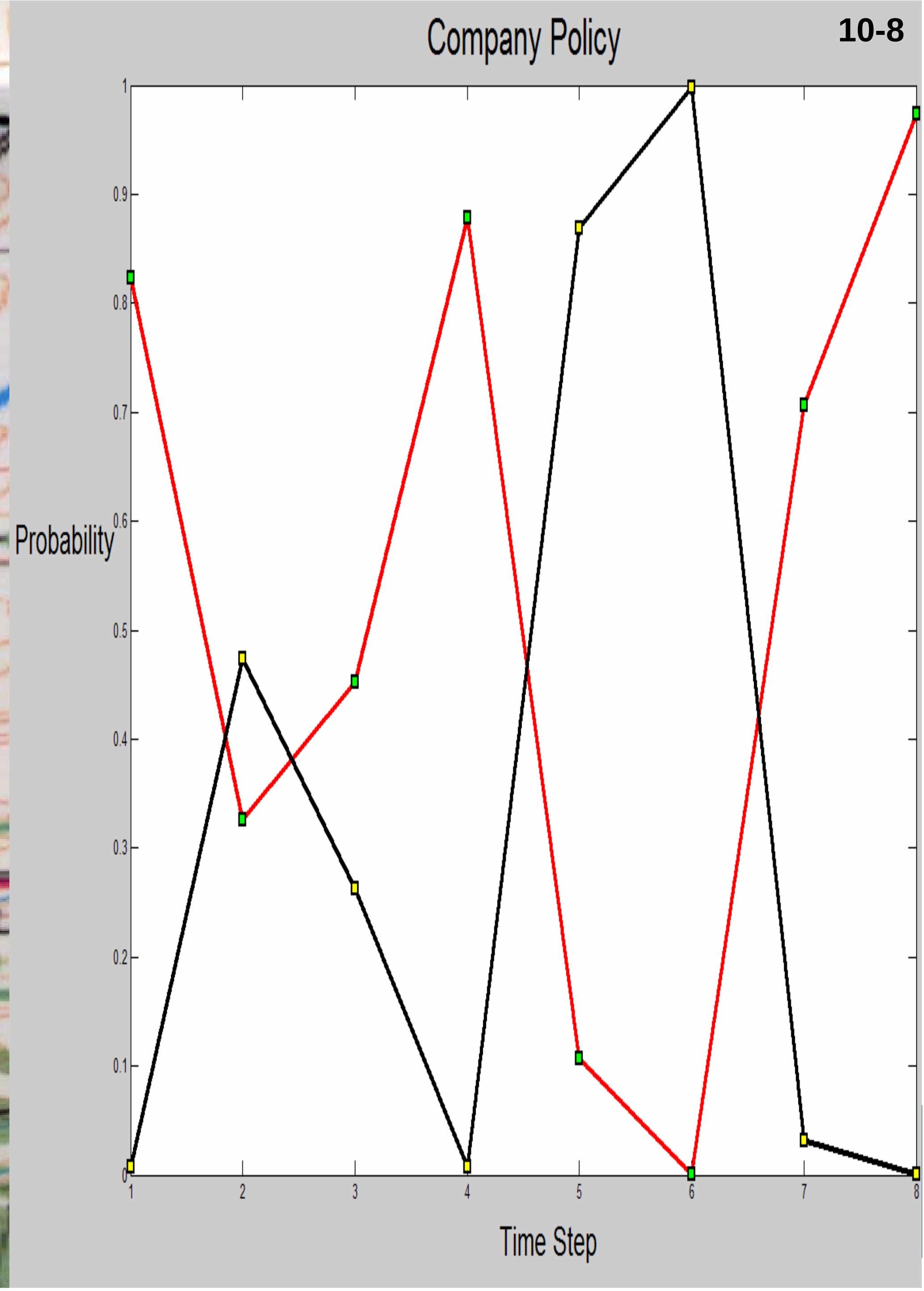
Results.exe

Smedstorp
Svesån
Risa
Helgesgården
Amundtorp
Hagen
Assartorp



Company Policy
10-7
Attack
March
Probability
Time Step


Smedstorp
Svesån
storp
Risa
Helgesgården
Amundtorp
Idatorp
Hagen
Assartorp


Company Policy
Probability
Time Step
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1
2
3
4
5
6
7
8

Contributions and Conclusions

- We have performed extension of the problem structure from previously studied single-agent policy recognition to multi-agent policy recognition that is applicable in military information fusion

Contributions and Conclusions

- Modeling agents and interrelations between agents
- A fragmented terrain representation is connected to policy recognition by soft evidence functions
- Implemented functions for recognition of the physical relations between the agents such as formations and enemy tank policies

Contributions and Conclusions

- We claim that with our model it is possible to integrate dynamical sensor data such as the enemy position and direction and to combine this knowledge with terrain data and uncertain a priori knowledge such as the doctrine knowledge to infer multi-agent policy in a robust and statistically sound manner.